IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	: Allan et al.	Art Unit	: 3693
Serial No.	: 09/696,099	Examiner	: Lalita M. Hamilton
Filed	: October 25, 2000	Conf. No.	: 4131
Title	: VALUE TRANSACTION SYSTEMS		

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

BRIEF ON APPEAL (REPLACEMENT)

(1) Real Party in Interest

The real party in interest is MEI, Inc., the current assignee of the application.

(2) Related Appeals and Interferences

None.

(3) Status of Claims

Claims 1-25 are pending and stand rejected under 35 U.S.C. § 103(a).

Claim 26 previously was canceled.

Applicant appeals the rejections of claims 1-25.

(4) Status of Amendments

All amendments have been entered.

(5) Summary of Claimed Subject Matter

The pending claims relate to value transaction systems in which code is uploaded from one or more transaction units and used to control operation of the same transaction unit(s).

1. Claims 1-6

Independent claim 1 recites a value transaction system including transaction units. Examples of transaction units include a coin changer 4, a banknote validator 6, a card reader 8 and a vending machine 10 (*see, e.g.*, FIG. 1).

The value transaction system also includes a controller that is operable to upload respective run-time interpreted code units from the transaction units. As explained in the Specification at page 2, lines 3-6, the term "code unit" can refer to a collection of software routines. An example of the controller is illustrated in FIGS. 1 and 2 and identified by reference numeral 16. The controller is operable to execute the code of each respective code unit and, in response to execution of the respective code unit, to generate signals to control the operation of the respective transaction unit.

In some implementations, such a system can provide the advantage that a new transaction unit of completely arbitrary type can be added to an existing transaction system and function correctly with the other units under the control of a central controller in which the software units are integrated to facilitate information exchange. That may be accomplished without requiring either (a) an on-line system with a central, remote software-storing server or (b) a system that incorporates high-powered "intelligent" transaction units (*see, e.g.*, Specification, page 2, line 21 - page 3, line 6).

Claims 2-6 depend directly or indirectly from claim 1.

2. Claims 7-13

Independent claim 7 recites a validation transaction unit including validator components enabling validation of a currency item and a microprocessor system. Examples of the validator components are components of the coin changer 4, which includes a system controller 16 with a microprocessor 18 (FIGS. 1 and 2).

The microprossor system includes a validation code unit operable to accept and process input signals from the validator components to validate the item of currency. The microprocessor system also includes a Java Virtual Machine, an example of which is shown in FIG. 2 as reference numeral 26. *See also* Specification at page 2, lines 16-20; page 3, lines 7-12; page 6, line 16 – page 8, line 6.

The microprocessor system further includes at least one Java application operable to perform controlling functions for a further transaction unit to which the validation transaction unit is connected. The microprocessor is operable to upload the Java application from the further

transaction unit. An example of the "further transaction unit" in the illustrated implementation is the banknote validator 6, the card reader 8 or the vending machine 10.

Claims 8-13 depend directly or indirectly from claim 7.

3. Claims 14-21

Independent claim 14 recites a transaction system that includes transaction units and a controller having a processor and memory means. Examples of transaction units are identified by reference numerals 4, 6, 8 and 10. An example of a controller is identified by reference numeral 16 (FIG. 1), which has a processor 18 and memory 20 through 36.

The controller is coupled to the transaction units and arranged to receive and send signals from and to the transaction units. The controller also is operable to upload from each transaction unit a respective code module containing executable code associated with that transaction unit for storage in the memory means. As described according to a particular example in the Specification (page 11, lines 11-14):

> [T]he units 6, 8, and 10 may each have a respective memory storing the bytecodes for the respective modules 30, 32 and 34. The contents of the memory in each unit are uploaded to the system controller 16 in an initialisation operation for that unit.

The controller is operable to execute the code in each respective code module. The code in each particular module is functional independently of the code in the other modules and performs processing operations in response to signals received from its respective transaction unit indicative of respective operations performed by that transaction unit (*see, e.g.*, Specification, page 7, lines 5-11). The code is further operable to cause the controller to generate controlling signals for sending to the respective transaction unit and capable of representing different functions to be performed by the transaction unit (*see, e.g.*, Specification, page 8, line 10 – page 11, line 2).

Claims 15-21 depend from claim 14.

4. Claims 22-24

Independent claim 22 recites a transaction system that includes a controller unit having a processor operable to execute instructions in Java code. An example of the controller unit is identified by reference numeral 16 in FIGS. 1 and 2. The controller unit 16 has a processor 18. As described in an example in the Specification (page 6, lines 6-9 and 14-15):

> The microprocessor 18 performs instructions under the control of a real-time operating system formed by code unit 20. The operating system 20 is a multi-tasking operating system which in this embodiment executes code in the code units 22, 24 and 26.
>
> * * *
>
> . . . The code unit 26 is a Java Virtual Machine.

The transaction system includes at least one transaction unit including means for performing value transactions under the control of the processor executing code uploaded from the transaction unit. As described in an example in the Specification (page 6, lines 18-20 and page 11, lines 11-14:

> High level operations of the transaction units 4, 6, 8 and [10] are performed by the Java Virtual Machine 26, under control of respective code units 28, 30, 32 and 34.
>
> . . .
>
> * * *
>
> [T]he units 6, 8, and 10 may each have a respective memory storing the bytecodes for the respective modules 30, 32 and 34. The contents of the memory in each unit are uploaded to the system controller 16 in an initialisation operation for that unit.

Claims 23-24 depend directly or indirectly from claim 22.

5. Claim 25

Independent claim 25 recites a method of assembling a transaction system where the system includes transaction units and a controller having a processor and memory means for storing executable code in respective code modules each associated with a respective one of the transaction units. The controller is operable to execute the code in each respective code module. Each code module is operable to cause the controller to generate controlling signals for sending to the respective transaction unit and capable of representing different functions to be performed by the transaction unit.

Examples of the transaction units are the banknote validator 6, the card reader 8 and the vending machine 10 (FIG. 1). Examples of the code modules are illustrated in FIG. 2 and identified by reference numerals 30, 32 and 34.

The method of claim 25 includes separately loading the executable code for the respective code modules from the associated transaction unit into the memory means of the controller. *See, e.g.*, Specification at page 11, lines 11-14; page 7, lines 5-22.

(6) Grounds of Rejection to be Reviewed on Appeal

1. Whether claims 1-6 were properly rejected under 35 U.S.C. §103(a) as unpatentable over U.S. Patent No. 6,311,165 (Coutts) in view of U.S. Published Patent Application No. 2001/0011680 (Soltesz et al.).

2. Whether claims 7-13 were properly rejected under 35 U.S.C. §103(a) as unpatentable over the Coutts patent in view of U.S. Patent No. 6,318,536 (Korman).

3. Whether claims 14-21 were properly rejected under 35 U.S.C. §103(a) as unpatentable over U.S. Patent No. 6,311,165 (Coutts) in view of U.S. Published Patent Application No.3 2001/0011680 (Soltesz et al.).

4. Whether claims 22-24 were properly rejected under 35 U.S.C. §103(a) as unpatentable over U.S. Patent No. 6,311,165 (Coutts) in view of U.S. Published Patent Application No. 2001/0011680 (Soltesz et al.).

5. Whether claim 25 was properly rejected under 35 U.S.C. §103(a) as unpatentable over U.S. Patent No. 6,311,165 (Coutts) in view of U.S. Published Patent Application No. 2001/0011680 (Soltesz et al.).

(7) Argument

The pending claims relate to value transaction systems in which code is uploaded from one or more transaction units and used, respectively, to control operation of the same transaction unit(s). In contrast, as discussed in greater detail below, in the cited references, software is downloaded or uploaded from one source (*e.g.*, a central server or the internet) and then used to control operation of some other component of the system.

1. Claims 1-6

Independent claim 1 recites a value transaction system including transaction units and a controller that is operable to upload *respective* run-time interpreted code units from the transaction units. The controller can execute the code of each *respective* code unit and generate signals to control the operation of the respective transaction units.

The Coutts patent relates to networked transaction systems having a central server and a terminal having a number of peripheral devices, such as a cash dispenser, card reader, etc. The systems disclosed by the Coutts patent have peripheral devices that download software from the central server and execute their own software. For example, the Coutts patent discloses:

> [T]he server is arranged to store applications and driver or other operational software for the peripheral devices and communication links can be provided from the server to individual peripheral devices to enable such software to be *downloaded from the server to the device.*
>
> * * *
>
> With the disclosed architecture, appropriate software can be readily *downloaded from server 16* through link 17 at run time without the need to store every alternative driver program at the dispenser.

(Col. 3, lines 58-62; col. 11, lines 6-9) (Emphasis added) Thus, the peripheral devices retrieve software from a source (*i.e.*, the server) that is external to the transaction system itself. As

explained by the Coutts et al. patent, such implementations "allow for the *remote* administration of an entire transaction network" (col. 9, lines 51-52).

Indeed, the Office action of September 22, 2006 acknowledges (at pages 2-3) that the Coutts patent "does not disclose the controller being operable to upload from the transaction units respective run-time interpreted code units for storing in the memory or separately loading executable code for the respective code modules from the associated transaction unit into the memory means of the controller." Nevertheless, the Office action cites the Soltesz et al. patent as allegedly disclosing those features. In particular, both the Office action of September 22, 2006 and March 29, 2007 refer to paragraphs 3 and 34 of the Soltesz et al. patent.

The Soltesz et al. patent discloses a self-service kiosk. Paragraphs 3 and 34 of the Soltesz et al. patent disclose connecting the kiosk to the Internet, for example "for downloading or uploading new programs." There is no indication, however, that the retrieved software is used for controlling the operations of respective transaction units within the kiosk. Moreover, the software is not retrieved from the transaction units.

Thus, the Coutts et al. patent merely discloses retrieval of code from a source external to the transaction system in the form of a central server. Likewise, the Soltesz et al. patent suggests retrieval of code from an external source (*i.e*, the Internet). Accordingly, the cited references neither individually nor in combination suggest the retrieval of code units from respective transaction units. In particular, even if there were some reason to combine the disclosures of the cited references, that would not result in or render obvious a controller "operable to upload from said transaction units respective run-time interpreted code units . . ., the controller being operable to execute the code of each respective code unit and in response thereto to generate signals controlling the operation of the respective transaction units," as recited in claim 1.

At least for those reasons, the rejections of claim 1 and dependent claims 2-6 should be reversed.

2. Claims 7-13

Independent claim 7 recites a validation transaction unit with a microprocessor system that includes at least one Java application operable to perform controlling functions for a further transaction unit to which the validation transaction unit is connected. The microprocessor is operable to upload the Java application from the further transaction unit.

The final Office action of March 29, 2007 does not even address claims 7-13 in any detail. Although the Office action of September 22, 2006 acknowledges that the Coutts patent does not disclose enabling validation of currency, the Office action alleges that otherwise "Coutts discloses the invention substantially as claimed." That is incorrect.

As discussed above, the systems disclosed by the Coutts patent have peripheral devices that download software from the central server and execute their own software. Therefore, there is no disclosure in the Coutts patent of a processor that is operable to upload an application from a transaction unit where the application is operable to perform controlling functions for that same transaction unit. Nor does the Korman et al. patent disclose that feature. The rejections of claims 7-13 should be reversed at least for these reasons.

Furthermore, the Office actions do not even address the fact that independent claim 7 specifically recites uploading a Java application from the further transaction unit. That feature, as well as the subject matter of claim 7 as a whole, is not suggested or otherwise rendered obvious by the combination of the Coutts et al. and Korman et al. patents.

At least for the foregoing reasons, the rejections of the claims 7-13 should be reversed.

3. Claims 14-21

Claim 14 recites that the controller is operable "to upload from each said transaction unit a respective code module containing executable code associated with that transaction unit" for storage in the memory means. The controller is operable to execute the code in each respective code module, and the code is "operable to cause the controller to generate controlling signals for sending to the respective transaction unit and capable of representing different functions to be performed by the transaction unit."

The Coutts et al. patent and the Soltesz et al. patent are discussed above in connection with the discussion of claim 1. The remarks regarding those patents are equally applicable here.

In particular, the Coutts et al. patent merely discloses that peripheral devices retrieve software from a source (*i.e.*, the server) that is external to the transaction system itself. Likewise, the Soltesz et al. patent suggests retrieval of code from an external source (*i.e*, the Internet). Accordingly, the cited references neither individually nor in combination suggest the retrieval of code units from respective transaction units. In particular, even if there were some reason to combine the disclosures of the cited references, that would not result in or render obvious the subject matter of claims 14-21.

At least for the foregoing reasons, the rejections of claim 14-21 should be reversed.

4. Claims 22-24

The transaction system of claim 22 includes a controller unit including a processor operable to execute instructions in Java code, and at least one transaction unit including means for "performing value transactions under the control of the processor executing code uploaded from the transaction unit."

The Coutts et al. patent and the Soltesz et al. patent are discussed above in connection with the discussion of claim 1. The remarks regarding those patents are equally applicable here.

In particular, the Coutts et al. patent merely discloses that peripheral devices retrieve software from a source (*i.e.*, the server) that is external to the transaction system itself. Likewise, the Soltesz et al. patent suggests retrieval of code from an external source (*i.e*, the Internet). Accordingly, the cited references neither individually nor in combination suggest the retrieval of code units from respective transaction units. In particular, even if there were some reason to combine the disclosures of the cited references, that would not result in or render obvious the subject matter of claims 22-24.

5. Claim 25

Claim 25 recites a method of assembling a transaction system that includes a controller operable to execute code in respective code modules, where each code module is operable to cause the controller to generate controlling signals for sending to the respective transaction unit and capable of representing different functions to be performed "by the transaction unit." The

method of claim 25 includes separately loading the executable code for the respective code modules "from the associated transaction unit into the memory means of the controller."

The Coutts et al. patent and the Soltesz et al. patent are discussed above in connection with the discussion of claim 1. The remarks regarding those patents are equally applicable here.

In particular, the Coutts et al. patent merely discloses that peripheral devices retrieve software from a source (*i.e.*, the server) that is external to the transaction system itself. Likewise, the Soltesz et al. patent suggests retrieval of code from an external source (*i.e*, the Internet). Accordingly, the cited references neither individually nor in combination suggest the retrieval of code units from respective transaction units. In particular, even if there were some reason to combine the disclosures of the cited references, that would not result in or render obvious the subject matter of claim 25.

At least for the foregoing reasons, the rejection of claim 25 should be reversed.

Conclusion

In view of the foregoing remarks, applicant respectfully requests reversal of all rejections and allowance of the pending claims.

It is believed that all of the pending claims have been addressed. However, the absence of a reply to a specific rejection, issue or comment does not signify agreement with or concession of that rejection, issue or comment. In addition, because the arguments made above may not be exhaustive, there may be reasons for patentability of any or all pending claims (or other claims) that have not been expressed. Finally, nothing in this paper should be construed as an intent to concede any issue with regard to any claim, except as specifically stated in this paper.

The brief fee of $510 is being paid electronically. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: 10/31/07

Samuel Borodach
Reg. No. 38,388

Fish & Richardson P.C.
Citigroup Center
52nd Floor
153 East 53rd Street
New York, New York 10022-4611
Telephone: (212) 765-5070
Facsimile: (212) 258-2291

30374613.doc

Appendix of Claims on Appeal

1. A value transaction system comprising a plurality of transaction units and a controller having a processor and memory means, the controller being operable to upload from said transaction units respective run-time interpreted code units for storing in said memory means, the controller being operable to execute the code of each respective code unit and in response thereto to generate signals controlling the operation of the respective transaction units.

2. A system as claimed in claim 1, further comprising a native code unit operable to accept and process input signals for the purpose of validation of an item of money.

3. A system as claimed in claim 1, wherein the transaction units are arranged to handle respective types of payment media.

4. A system as claimed in claim 1, wherein each interpreted code unit is independently functional without regard to the presence of the other interpreted code units.

5. A system as claimed in claim 4, including an API code unit containing routines which are accessible at run-time by each of the interpreted code modules.

6. A system as claimed in claim 1, wherein the memory means is a non-volatile semiconductor memory.

7. A validation transaction unit for a value transaction system, the validation transaction unit comprising validator components enabling validation of a currency item and a microprocessor system including:

(a) a validation code unit operable to accept and process input signals from said validator components for the purposes of validation of said item of currency;

(b) a Java Virtual Machine; and

(c) at least one Java application operable to perform controlling functions for a respective further transaction unit to which the validation transaction unit is connected,

wherein the microprocessor system is operable to upload the Java application from the further transaction unit.

8. A validation transaction unit as claimed in claim 7, wherein the validation code unit comprises native code.

9. A validation transaction unit as claimed in claim 7, wherein the validation code unit comprises compiled code.

10. A validation transaction unit as claimed in claim 7, including a further Java application operable to perform controlling functions for the validation transaction unit.

11. A validation transaction unit as claimed in claim 7, wherein the validation transaction unit is a coin validation mechanism.

12. A transaction system comprising a validation transaction unit as claimed in claim 7, and at least one further transaction unit under the control of the microprocessor system in said validation transaction unit.

13. A transaction system as claimed in claim 12, wherein the transaction units are interconnected via a serial link.

14. A transaction system comprising:

a plurality of transaction units; and

a controller having a processor and memory means, the controller being coupled to the transaction units and arranged to receive and send signals from and to the transaction units, the controller being operable to upload from each said transaction unit a respective code module containing executable code associated with that transaction unit for storage in said memory means;

the controller being operable to execute the code in each respective code module, the code in that module being functional independently of the code in the other modules and performing processing operations in response to signals received from its respective transaction unit indicative of respective operations performed by that transaction unit, and the code being further operable to cause the controller to generate controlling signals for sending to the respective transaction unit and capable of representing different functions to be performed by the transaction unit.

15. A transaction system as claimed in claim 14, wherein the memory means has executable code in a further code module, that executable code being responsive to credit-representing signals generated by the code in one or more other code modules, and being operable to produce vend-authorising signals for use by the executable code in at least one other code module.

16. A transaction system as claimed in claim 14, wherein the executable code is run-time interpreted code.

17. A transaction system as claimed in claim 14, wherein the controller is housed in one of the transaction units.

18. A transaction system as claimed in claim 14, wherein each code module is contained in a respective area of protected memory.

19. A transaction system as claimed in claim 14, wherein the executable code is Java bytecode.

20. A transaction system as claimed in claim 14, wherein the transaction units are interconnected via a serial link.

21. A transaction system as claimed in claim 14, wherein the transaction units include one or more of (a) a coin mechanism unit, (b) a banknote mechanism unit, (c) a card reader unit and (d) a vending machine controller unit.

22. A transaction system comprising a controller unit including a processor operable to execute instructions in Java code, and at least one transaction unit including means for performing value transactions under the control of the processor executing code uploaded from the transaction unit.

23. A transaction system as claimed in claim 22, wherein the transaction system comprises a plurality of transaction units, and the controller unit is operable to execute code stored in respective code units each associated with a respective transaction unit.

24. A transaction system as claimed in claim 23, wherein the code units are stored in respective protected memory areas.

25. A method of assembling a transaction system, the transaction system comprising a plurality of transaction units and a controller having a processor and memory means for storing executable code in respective code modules each associated with a respective one of the transaction units, the controller being coupled to the transaction units and arranged to receive and send signals from and to the transaction units, and the controller being operable to execute the code in each respective code module, each code module performing processing operations in response to signals received from the respective transaction unit indicative of respective operations performed by that transaction unit, and the code module being further operable to cause the controller to generate controlling signals for sending to the respective transaction unit and capable of representing different functions to be performed by the transaction unit; the method comprising:

separately loading the executable code for the respective code modules from the associated transaction unit into the memory means of the controller.

Evidence Appendix

None.

Related Proceedings Appendix

None.

30374613.DOC